LHSG-CT-2003-503420

BioXHIT

A European integrated project to provide a highly effective technology platform for Structural Genomics.

Life Sciences, Genomics and Biotechnology for Health

WP5.2: De 5.2.17 Reaper: extension of starKey application for structure deposition

Due date of deliverable: **31.03.2008**
Actual submission date: **30.06.2008**

Start date of project: **1.1.2004**   **Duration: 60 months**

Organisation name of lead contractor for this deliverable: CCP4/STFC Daresbury **Author:** Peter Briggs

De 5.2.17 Reaper: extension of starKey application for structure deposition

A new program called "reaper" has been developed which uses the concepts and underlying code previously developed in the starKey program (deliverable 5.1.4).

Like starKey, Reaper uses the Python client API to interact with the project tracking database and performs an analysis of the job history plus the logfiles associated with the jobs in order to suggest files and data that could be "harvested" for deposition.

Specifically, given a project database name and an initial job id, Reaper tracks backwards and identifies all the jobs in the database that led up to that job – essentially generating a "path" through the project history o the initial job. Reaper then analyses each job in the path and attempts to assign it to a "harvesting stage", which is similar to the harvesting stages recognised by deposition software such as PDB_EXTRACT, from the RCSB-PDB. Where more than one job in the path can be assigned to a single harvesting stage, only the most recent job is actually assigned.

Reaper is a Python program and can be run in a manual mode directly from the command line using the syntax

```
> python reaper.py <project> <jobid>
```

In this case the analysis is written out to the terminal window (see example shown in figure 1).

Figure 1: ouptut of Reaper program run from Linux command line.

Reaper has also been incorporated into the "dbviewer" visualiser application (developed for deliverable 5.2.7), which provides a more intuitive way for most people to use the program.

Right-clicking on a job in the visualiser gives a menu of options, one of which is "Run Reaper for Data Harvesting" (see figures 2 and 3).
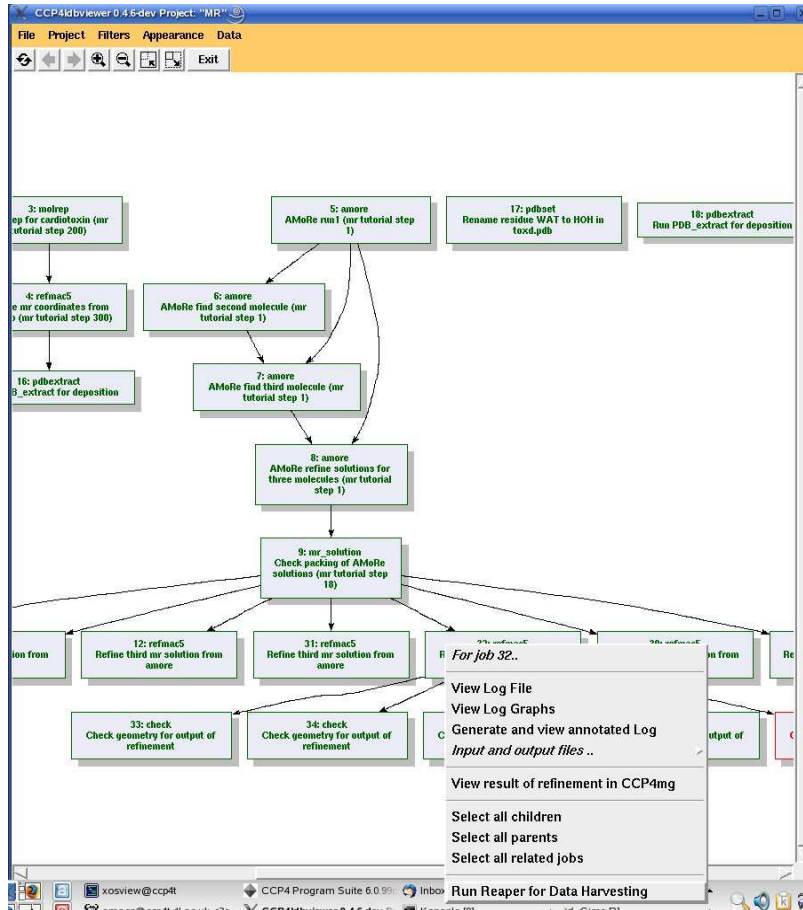

Figure 2: example showing context menu when user right-clicks on a job; the last option is to "run Reaper for Data Harvesting".

Selecting this option, the viewer then invokes Reaper in background using the current project and selected job as input. The view is then updated to show only the jobs in the path identified by Reaper (see figure 4), and a window pops up listing each of the harvesting stages with details of the job (if any was found) assigned to it, including a list of the associated output files (see figure 5).

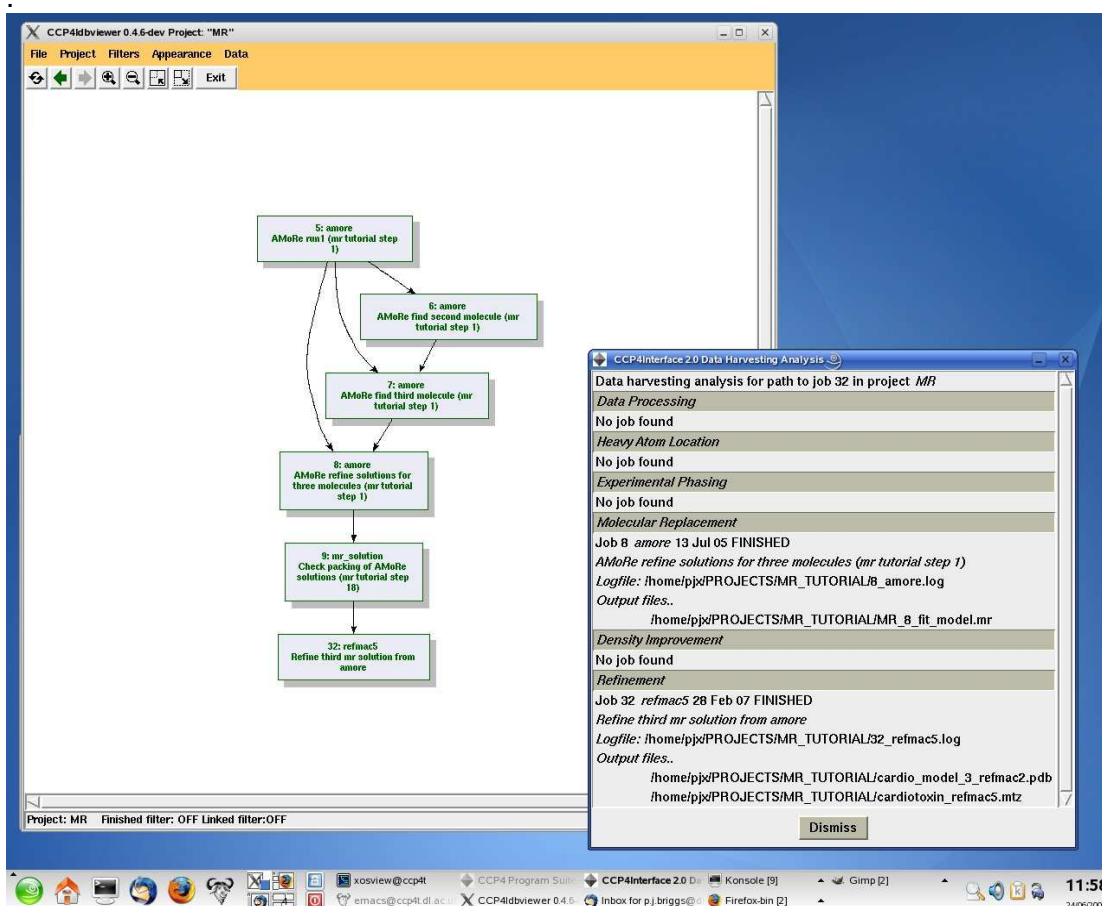**Figure 3: close up view of the context menu in the dbviewer from figure 2.**

Figure 4: output after running Reaper from dbviewer: the view updates to show only the jobs in the "path" leading to the initial job, and a separate window shows details of the job associated with each harvesting stage.
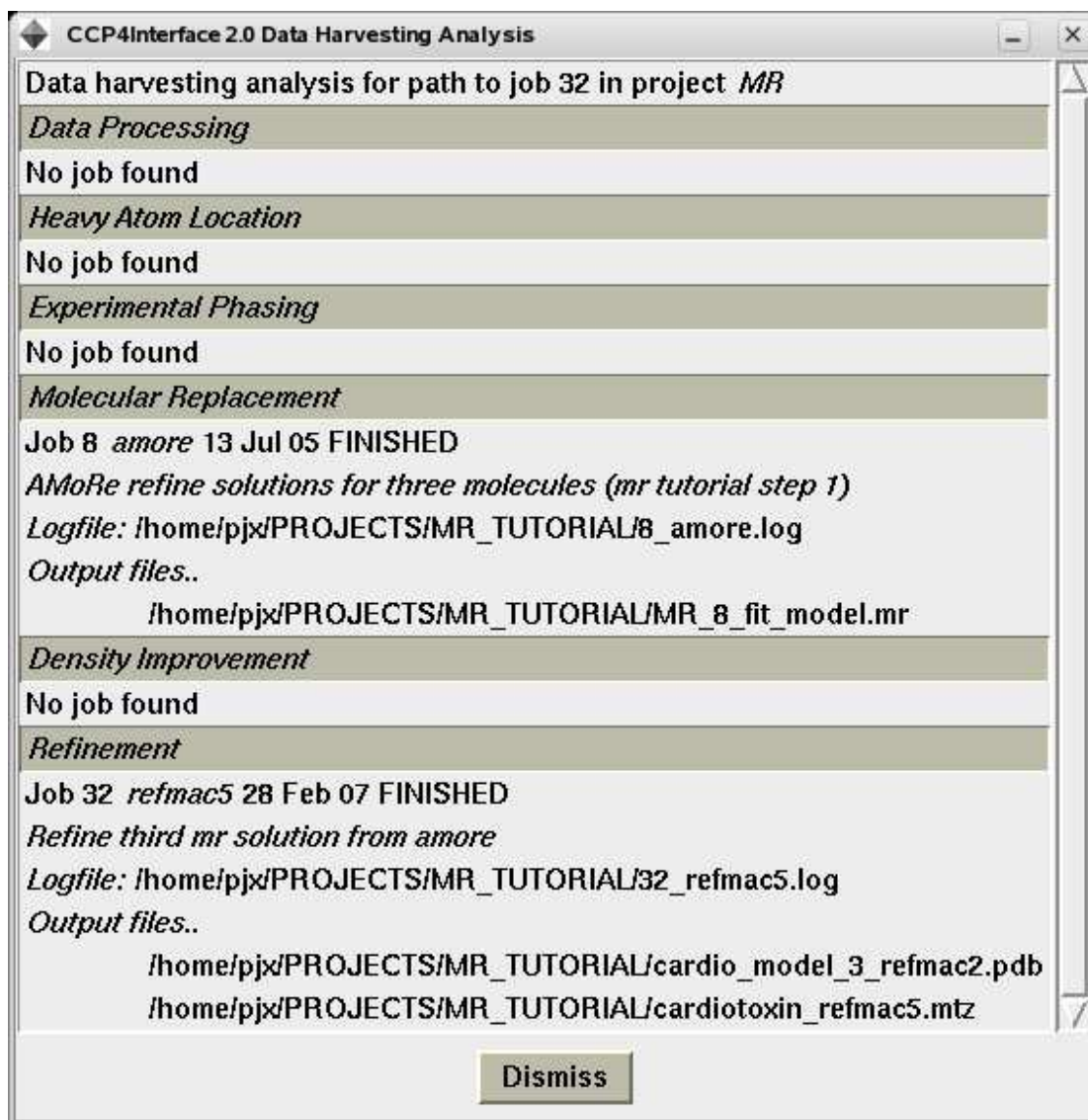
Figure 5: close up view of the window showing the harvest stages generated by Reaper in figure 4.

Reaper is provided with dbccp4i 0.4.6, which can be found in the following supplementary files at http://www.ccp4.ac.uk/projects/bioxhit_public:

- de5.2.17_dbccp4i-0.4.6.tar.gz
- de5.2.17_dbccp4i-0.4.6.zip

Peter Briggs, CCP4